

# Intro: Url Endpoints used

App menu tree:

Main menu - <https://mylinks.aghayev.com/api>

Sub menu: <https://mylinks.aghayev.com/api/subcategories>

Menu page: <https://mylinks.aghayev.com/api/weblinks/{slug}>

# Intro: App folder structure

App router:

app/

Navbar.js – ISR, fetch { revalidate: 10 }

weblinks/[slug]/page.tsx –

SSG – generateStaticParams

protected/\*

services/\*

lib/\*

# Navbar.js – ISR

ISR – Incremental Static Regeneration.

Used if page content depends on external data

App Router:

```
const res = await fetch('https://.../posts',{ next: { revalidate: 10 } })
```

Pages Router:

```
export async function getStaticProps() {  
  const res = await fetch('https://.../posts')  
  const posts = await res.json()  
  return { props: { posts }, revalidate: 10 }  
}
```

npm run build

Result: If client-side data fetch - static site data stale for 10 seconds, then re-fetches.

If SSR and revalidate used - data re-fetched every 10 seconds

# weblinks/[slug]/page.tsx - SSG

## SSG – Static Site Generation

### Used for statically generate routes

App router, app/weblinks/[slug]/page.tsx:

```
export async function generateStaticParams() {  
  const res = await fetch('https://.../subcategories')  
  const categories = await res.json()  
  return categories.map(({ slug }) => ({  
    slug: slug,  
  })))
```

Pages router:

getStaticPaths()

npm run build

● /weblinks/[slug]	182 B	87.4 kB
├ /weblinks/general		
├ /weblinks/finance		
├ /weblinks/my-livejournal		
└ [+13 more paths]		

● (SSG) prerendered as static HTML (uses `getStaticProps`)

# App Router file conversion loading.js

```
app/weblinks/[slug]/loading.tsx  
export default function Loading() {  
  return <p>Loading...</p>  
}
```

# Capacitor Installation, page 1

package.json

```
"static": "next build && next export"
```

next.config.mjs

```
const nextConfig = { output: 'export' }
```

```
export default nextConfig
```

capacitor.config.js

```
... "webDir": "out"
```

# Capacitor installation, page 2

Installation:

```
npm i @capacitor/core
```

```
npm i -D @capacitor/cli
```

```
npm cap init
```

```
npm i @capacitor/android @capacitor/ios
```

Build time:

```
npm run build
```

```
npm cap add ios (npm cap add android)
```

```
npm cap sync ( next.js codebase translation to ios/ or android/ codebase) – if fails for ios/ - see Capacitor Installation 3
```

```
IOS: Code copied to: /ios/App/App/public
```

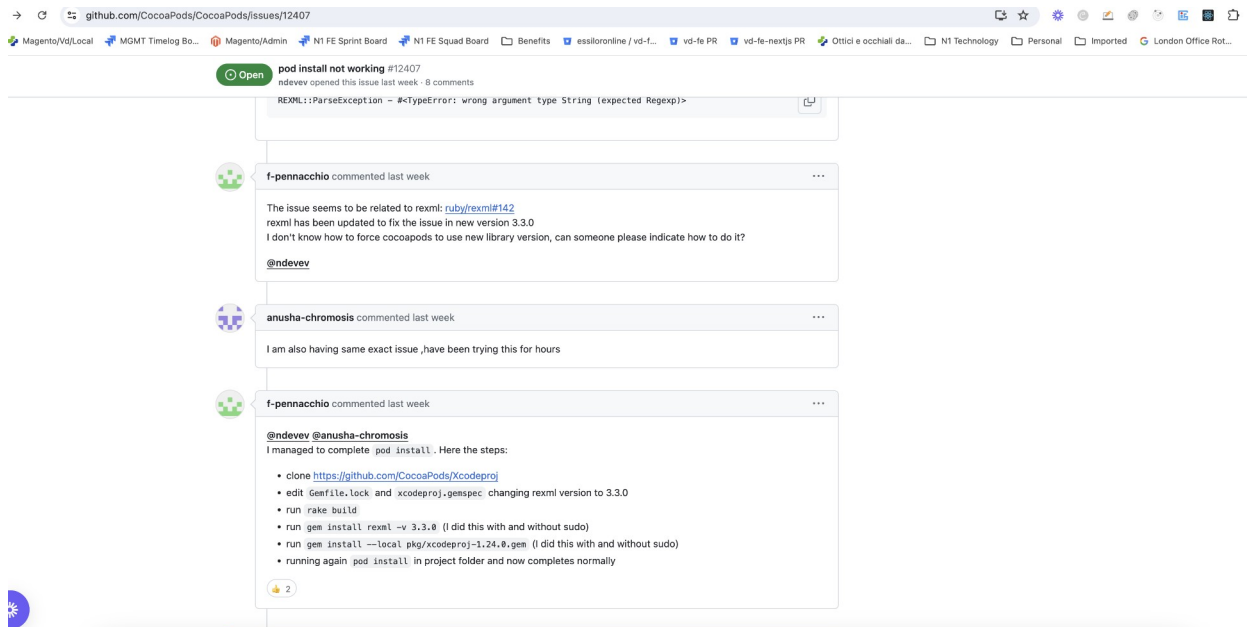
```
Android: Code copied to: /android/app/src/main/assets/public
```

```
npm cap open ios (or npm cap open android – opens Xcode IDE or Android Studio)
```

# Capacitor Installation, page 3

For ios/ on some Macs build fails

To fix: follow this: [Xcodeproj CocoaPods](https://github.com/CocoaPods/CocoaPods/issues/12407)



The screenshot shows a GitHub issue titled "pod install not working #12407" on the CocoaPods repository. The issue was opened by ndevex and has 8 comments. The error message is: `REXML::ParseException - #<TypeError: wrong argument type String (expected Regexp)>`. The issue is related to the `rexml` gem, which has been updated to version 3.3.0. The user is asking for instructions on how to force CocoaPods to use the new library version.

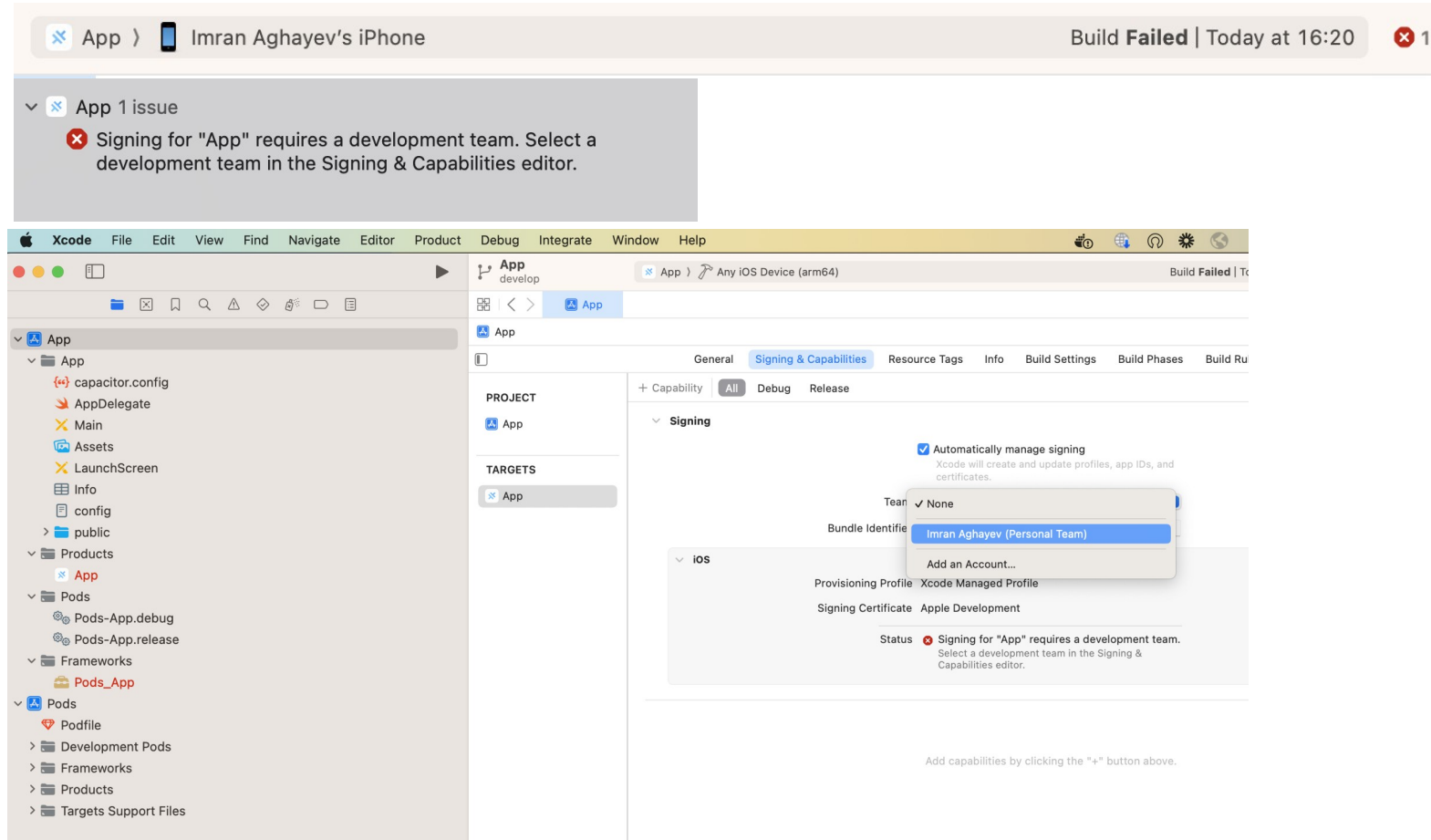
**Comments:**

- f-pennacchio** commented last week: The issue seems to be related to rexml: [ruby/rexml#142](#). rexml has been updated to fix the issue in new version 3.3.0. I don't know how to force cocoapods to use new library version, can someone please indicate how to do it? @ndevex
- anusha-chromosis** commented last week: I am also having same exact issue ,have been trying this for hours
- f-pennacchio** commented last week: @ndevex @anusha-chromosis I managed to complete pod install. Here the steps:
  - clone <https://github.com/CocoaPods/Xcodeproj>
  - edit Gemfile, lock and Xcodeproj.gemspec: changing rexml version to 3.3.0
  - run rake build
  - run gem install rexml --v 3.3.0 (I did this with and without sudo)
  - run gem install --local pkg/xcodeproj-1.24.8.gem (I did this with and without sudo)
  - running again pod install in project folder and now completes normally

2



# Deploying to iOS, page 1



App > Imran Aghayev's iPhone Build **Failed** | Today at 16:20 1

App 1 issue

✖ Signing for "App" requires a development team. Select a development team in the Signing & Capabilities editor.

Xcode File Edit View Find Navigate Editor Product Debug Integrate Window Help

App develop App > Any iOS Device (arm64) Build Failed | Today at 16:20

App

App

capacitor.config AppDelegate Main Assets LaunchScreen Info config public Products App Pods Pods-App.debug Pods-App.release Frameworks Pods\_App Pods Podfile Development Pods Frameworks Products Targets Support Files

PROJECT

App

TARGETS

App

General Signing & Capabilities Resource Tags Info Build Settings Build Phases Build Rules

+ Capability All Debug Release

Signing

Automatically manage signing Xcode will create and update profiles, app IDs, and certificates.

Team

Bundle Identifier

Imran Aghayev (Personal Team)

Add an Account...

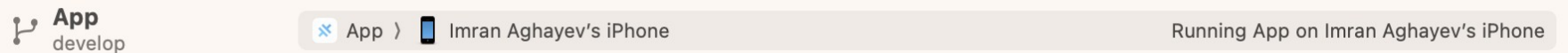
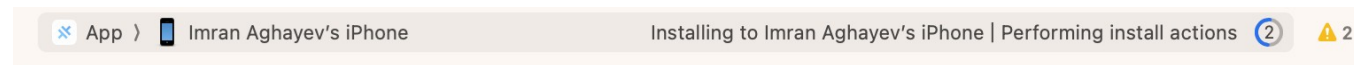
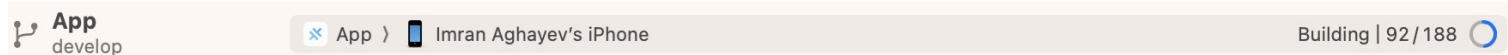
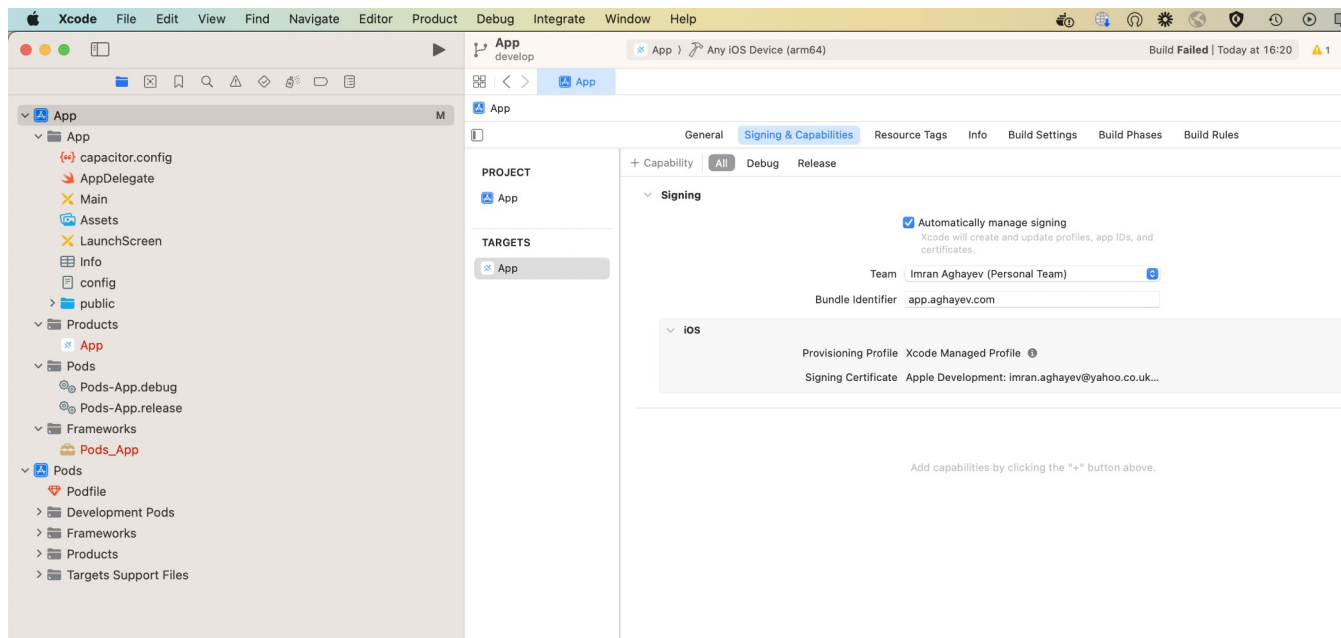
Provisioning Profile Xcode Managed Profile

Signing Certificate Apple Development

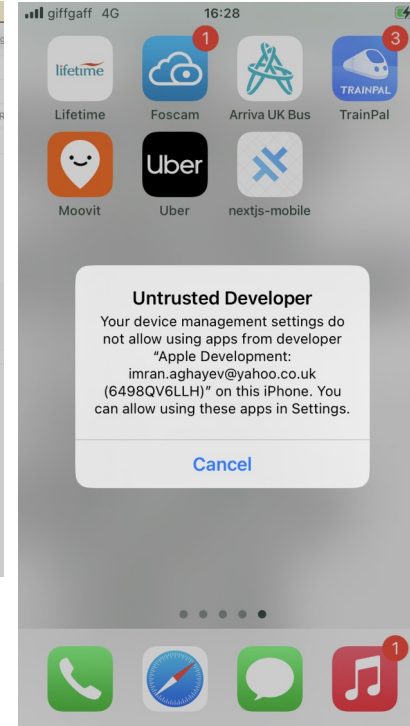
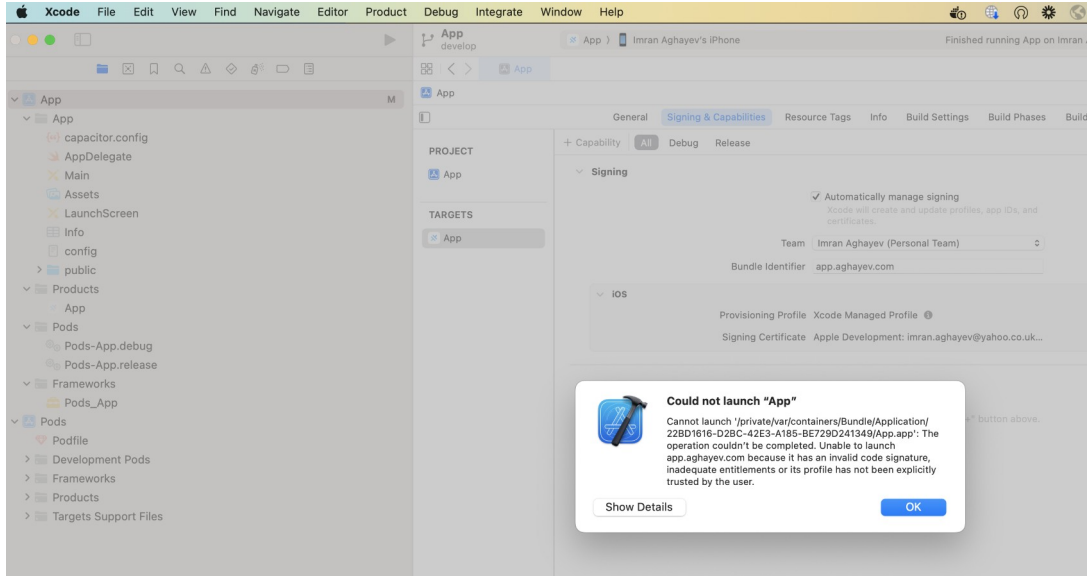
Status ✖ Signing for "App" requires a development team. Select a development team in the Signing & Capabilities editor.

Add capabilities by clicking the "+" button above.

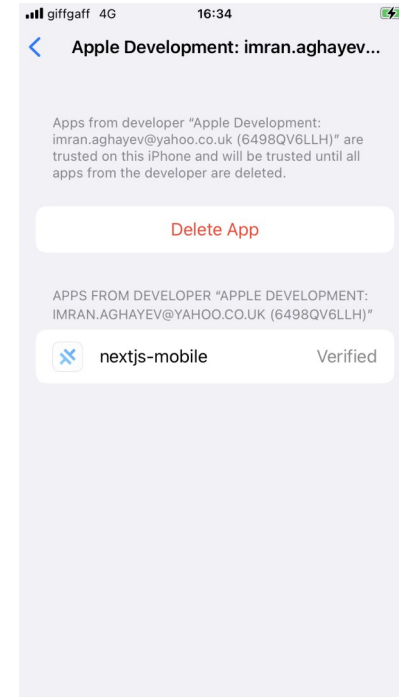
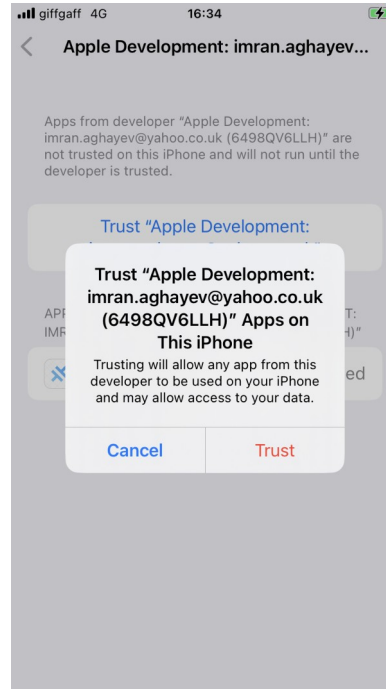
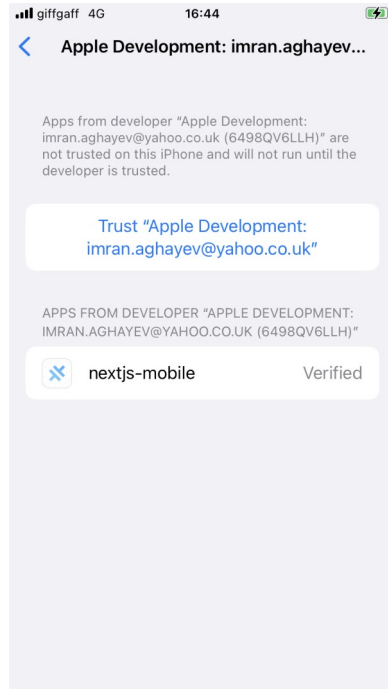
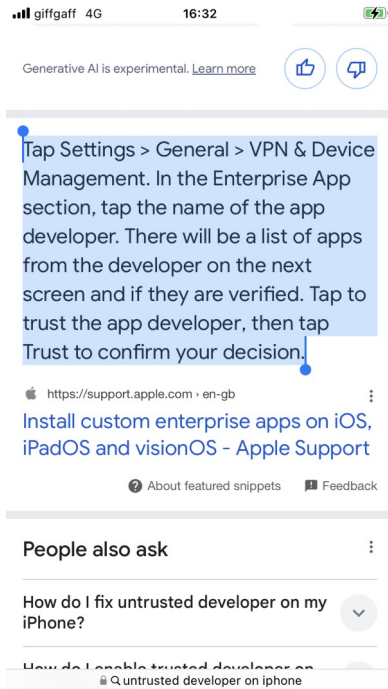
# Deploying to iOS, page 2



# Deploying to iOS, page 3



# Deploying to iOS, page 4



# Resources:

1. [Next.Js Mobile App in minutes with Capacitor](#)
2. [Capacitor Installation Instruction](#)
3. [Capacitor official documentation](#)
4. [Adding Signing Capabilities to your IOS App](#)
5. [Deploying to Android phone](#)
5. [Next.Js SSG and ISR](#)
6. [Next.Js App Router Migration](#)